Event Detection and User Interest Discovering in Social Media Data Streams

Lei-lei Shi[1], Lu Liu[1,2], Yan Wu[1], Liang Jiang[1], James Hardy[2]
[1]School of Computer Science and Telecommunication Engineering, Jiangsu University, China
[2]Department of Computing and Mathematics, University of Derby, UK
Email: l.liu@derby.ac.uk

Abstract—Social media plays an increasingly important role in people's life. Microblogging is a form of social media which allows people to share and disseminate real-life events. Broadcasting events in microblogging networks can be an effective method of creating awareness, divulging important information and so on. However, many existing approaches at dissecting the information content primarily discuss the event detection model and ignore the user interest which can be discovered during event evolution. This leads to difficulty in tracking the most important events as they evolve including identifying the influential spreaders. There is further complication given that the influential spreaders interests will also change during event evolution. The influential spreaders play a key role in event evolution and this has been largely ignored in traditional event detection methods. To this end, we propose a user-interest model based event evolution model, named the HEE (Hot Event Evolution) model. This model not only considers the user interest distribution, but also uses the short text data in the social network to model the posts and the recommend methods to discovering the user interests. This can resolve the problem of data sparsity, as exemplified by many existing event detection methods, and improve the accuracy of event detection. A hot event automatic filtering algorithm is initially applied to remove the influence of general events, improving the quality and efficiency of mining the event. Then an automatic topic clustering algorithm is applied to arrange the short texts into clusters with similar topics. An improved user-interest model is proposed to combine the short texts of each cluster into a long text document simplifying the determination of the overall topic in relation to the interest distribution of each user during the evolution of important events. Finally a novel cosine measure based event similarity detection method is used to assess correlation between events thereby detecting the process of event evolution. The experimental results on a real Twitter dataset demonstrate the efficiency and accuracy of our proposed model for both event detection and user interest discovery during the evolution of hot events.

Keywords- Microblogging; Event Evolution; User Topic;

I. INTRODUCTION

In recent years, social media such as Twitter, Facebook and Sina Weibo [1] are becoming an indispensable part in our daily lives. The rapid development of social media has attracted a large amount of research attention [2-8]. Microblogging is a form of social media where a user shares status updates and opinions in the form of short messages. This is known as a "tweet" in Twitter and has a maximum limit of 140 new characters per message. The recent popularity of microblogging networks clearly shows that microblogging is continuing to develop and rapidly attract people. In general, the information and data being posted in microblogging is most often event-driven [2]. This has allowed microblogging to become an important source for reporting real-world events.

A real-world occurrence reported in microblogging is normally referred to as a social event which may hold critical information describes events and circumstances that exist during a crisis [3]. In real applications, such as crisis management and opinion control, monitoring critical events over social media streams may enable support service officers to analyse all aspects of a composite event. This enables the most informed decisions to be made based on the detailed contexts such as the nature of the composite event, where the individual events are taking place and who the participants are. Moreover, ease of use as an information sharing platform has allowed microblogging to attract numerous users who generate a large number of events on a daily basis. Given the volume of data being generated, the potential benefits can only be realised if an information base regarding the salient events and influential spreaders is maintained.

Most existing research is limited to event detection methods and ignores the event evolution. This leads to difficulty in tracking the most important events as they evolve including identifying the influential spreaders. There is further complication given that the influential spreaders interests will change during event evolution leading to problems of low search efficiency and low accuracy of results. Detecting the event evolution relationship is necessary to formulate a deep understanding of the main development trend for the topic and precisely locate the significant events along with the topic evolution.

Significant events, also known as hot events, detected by most existing methods are presented in the form of multiple keywords. The identification, classification and annotation of hot events are performed manually, greatly reducing the efficiency of the hot event detection. Moreover, most of the existing topic models directly applied to microblogging will encounter problems with data sparsity, resulting in the low quality identification of the main topic of the hot event and the interest of each user.

To solve the above problems, we propose a user-interest model based event evolution model, named the HEE (Hot Event Evolution) model. This model not only considers the changes in the interests of the user during the evolution of hot events, but also uses the short text data in the microblogging network to model the data. Methods to discover the user interest can then be recommended,

resolving the problem of data sparsity. This in turn improves the quality of topic definition of hot events and the interest of each user. Specifically, a hot event automatic filtering algorithm is proposed to remove the influence of general events, improving the quality and efficiency of mining event. An automatic topic clustering algorithm is proposed to combine all short texts with similar topics into clusters. An improved user-interest model is proposed to integrate all short texts in each cluster to form a long text document simplifying the determination of the overall topic in relation to the interest distribution of each user during the evolution of hot events. This addresses the problem of sparse data and improves the quality of topic definition. Finally, a cosine measure based event similarity detection method is proposed to judge correlation between events, which can detect the process of event evolution. The experimental results on Twitter dataset demonstrate the efficiency and accuracy of our proposed model.

The main contributions of this paper are listed as follows:

1) We propose an improved user-interest model based event evolution model, named HEE (Hot Event Evolution) model. This model not only considers the user's interest distribution, but also uses the short text data in microblogging network to model the data, addressing the problem of data sparsity, improving the quality of the topic definition and the interest level of each user during the evolution of hot events. An automatic hot event filter [19] is used rank popularity, removing the influence of general events and improving the quality of mining event. A topic clustering algorithm is then used to collate the related short texts into a single text document to solve the problem of sparse data. Finally, according to the users in the document and the scored topics, the topics of each document are modelled by LDA (Latent Dirichlet Allocation) [4] topic model to get the topics of the whole document and the interests of users.
2) We propose a novel cosine measure based event similarity detection method. The user interest communities are discovered through community detection methods [5], extracting the hottest event each user belongs to. Posts and users in each topic take different weights to describe their importance, improving the efficiency and accuracy of HEE model. A long text document is composed from posts in each of the interest communities to get the proper number of key keywords in each topic that are chosen as major event. Finally, the included angle cosine is used as the standard to judge correlation between events, which significantly improves the efficiency and accuracy of event evolution.
3) Experimental results on microblogging networks clearly show that the HEE model could provide richer information for the community structure of the detected events, and indicate the efficiency and accuracy of our proposed model for both event detection and user interest discovering during the evolution of hot events.

The remainder of this paper is structured as follows: we discuss related work for event detection and event evolution in Twitter in Section 2. Then, in Section 3, we introduce our HEE model. Finally, we present our experiments in Section 4. And the last section concludes our study.

II. Related Work

Event detection methods based on topic modelling are increasing in popularity. For example, PLSA (Probabilistic Latent Semantic Analysis) [6] and LDA are two important approaches to detect the hidden variables in microblogging. Such methods model the word occurrences with a probabilistic theory, and measure the topical similarity among the words. Although there has been many researches on detecting a target event in microblogging networks, a large amount of the existing research [2-4,6-8] only discusses event detection methods but ignore the event evolution and strategies to combine messages regarding the same event. This leads to problems in dynamically tracking hot events and with the identification of the influential spreaders. When information relating to the same event is not correctly synchronised, problems of low efficiency and low accuracy may result. Moreover, in a crisis situation, we often want to analyse the key users related to events. Problems associated with detecting the same events and integrating ambiguous views from different users may arise.

Several research models have been proposed, the key being determining how to represent an event. Different representation models will directly influence the accuracy of the similarity calculation between events. Some existing research [10-11] used the traditional vector space model (VSM) [9] to represent an event. Makkonen et al. [10] build a multi-vector event model to describe events and evaluate event relationships by computing the similarity of these vectors. The advantages of VSM-based representation methods are simplicity and intuitive graphical representation of events and their similarities. However, it treats the different vectors equally and does not distinguish the value of contribution from keywords, locations, posts, topics and users. In point of fact, each of these attributes from the event are related to each other and several of them are combined together to express various semantics. To address this, Nallapati et al. [11] test different combinations of attributes to and propose that the best choice is obtained from average content similarity combined with time decay. While the method is highly suited when applied to mass media news events, it cannot be applied directly to short texts.

The mining of user interests in microblogging networks are mainly based on topic models, a type of unsupervised generative probabilistic modelling. The process simulates the generation of a single document and then, using parameter estimation, determines the probable theme. However, the traditional topic models [4,6] are not effective when applied to user interest discovery. The main reasons for this are that

microblogging text is short, contains a high level of colloquial language and rarely provides context information for reference. Modelling and analysis of short texts will be limited by problems of sparse data features which results in lower quality of topic mining.

There are presently two solutions to address the data sparsity problem. The first method utilises a large number of external resources to complement the short text expression [12-13]. This method can have an over reliance on external resources; the addition of large numbers of external resources may significantly alter the original text semantics, giving rise to inaccurate results. The second method is to combine all short texts related to an event into a single long text documents and then apply topic modelling [14-15] to detect the user interests. The decision to integrate short texts into the long document is based on common characteristics of the short texts such as common keywords. To some extent this method resolved the problem of sparse data; however it did not significantly improve the quality of the mining topic. Yan et al. [16] proposed a new kind of bilateral word topic model, known as the BTM model. This directly analyses the entire document for word co-occurrence patterns and is an effective way to solve the problem of sparse feature of short text. However the BTM model cannot analyse the distribution of the topic against individual users and there can be scaling problems when analysing the bilateral words in the large documents. This can lead to a low quality of mining topic result especially when the complexity of word semantics is taken into account. Yali et al. [17] improved the BTM model, and put forward a Dirichlet process based BTM model to excavate the short text topic. This improvement determines for the number of topics automatically, the quality of the mining topic has not been improved. Individual users are an important aspect of a microblogging network. Topic information is generated by the users and therefore an improved analysis at the user level may lead to a better understanding the topic of microblogging networks. Discovery of underlying interests for an individual user is achieved by mining latent topics from large numbers of microblogging texts that they have published. Moreover, Rosen et al. [18] proposed an author topic model known as ATM, which can get the author's topic distribution. However this model is only suitable for long texts and is not applicable to short text modelling. Zhao et al. [15] proposed a Twitter-LDA topic model. This also considered user interest information but the model is based on external resources to model the text and generates results with low accuracy is discussed previously.

In this paper, we design a user-interest model based event evolution model, named the HEE model. Although previous research has applied models to event detection and user interest discovery during event evolution, our work is very different. This model not only considers the changes in the interests of the user during the evolution of hot events, but also uses the short text data in the microblogging network to model the data. Methods to discover the user interest can then be recommended, resolving the problem of data sparsity. This in turn improves the quality of topic definition of hot events and the interest of each user. Our test dataset is extracted from Twitter. We validate our model on the dataset for both event detection and user interest discovering during the evolution of hot events, comparing it with existing models.

III. HEE MODEL

The HEE model is composed of four modules: Specifically, first, a hot event automatic filtering algorithm is proposed to remove the influence of general events. Then, an automatic topic clustering algorithm is proposed to combine all short texts with similar topics into clusters. And an improved user-interest model is proposed to integrate all short texts in each cluster to form a long text document simplifying the determination of the overall topic in relation to the interest distribution of each user during the evolution of hot events. Finally, a cosine measure based event similarity detection method is proposed to judge correlation between events, detecting the process of event evolution.

A. Preliminary

We collect tweets during a certain period of time. However, an event's influence may change over time. Hence, we concentrate on analysing the influence of the tweet toward hot topics. In terms of tweets, we only achieve those which have a reply or retweet relationship to improve the efficiency and accuracy of hot event detection and user interest discovery during event evolution.

Definition 1 (Tweet) A tweet P is defined as a 4-tuple, P = (*id, text, user, time*), where *P.id* is a unique numerical identifier associated with the tweet, *P.text* is its textual content, *P.user* and *P.time* denote the user who posted and the creation time of the tweet P, respectively.

Definition 2 (Event) An event E is defined as a 5-tuple, E= (*time, location, key posts, topics, key users*), where *time* is the timestamp when an event was detected, *location* is the location where the event happened, *key posts* are a set of microbloggings that are related to this event, and has size limitation of 5, *topics* are the narrative description and summarization of the event, which is usually a short sentence, *key users* are a set of participants who were involved in this event.

Given a latest event E_0, we define the evolution chain of E_0 as a sequential pattern:

$$E_n \rightarrow E_{n-1} \rightarrow ... \rightarrow E_1 \rightarrow E_0 \quad (1)$$

where $E_i \rightarrow E_j$ means event developed from E_i to E_j . This chain traces back the whole development of the hot event E_0 along the timeline.

Definition 3 (User Profile) Let T be a specified time interval, an Interest Profile of user u in time interval T , called P_u^T , is represented by a vector of tweets ($P_{u,1}$, $P_{u,2}$, ..., $P_{u,K}$). Each component $P_{u,n}$ of P_u^T denotes the degree of u's interest in the topic $z_n \in \{z_1, z_2, ..., z_K\}$ in time interval T .

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2017.2675839, IEEE Access

In our work, a profile for a user is a collection of tweets showing the interest of a given user with regards to the available topics in the microblogging network in *T*.

B. An Event and User Filtering Method

In the HEE model, an event and user automatic filtering method is used to detect the number of events and rank them according to popularity.

Fig. 1 illustrates the process involved in event and user filtering.

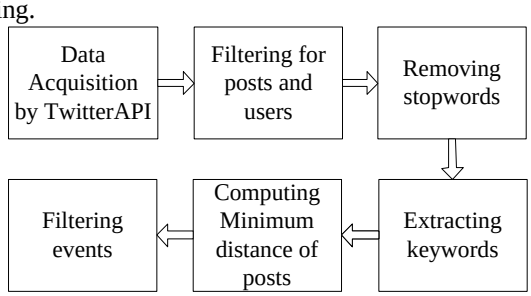


Figure 1. The Procedure of event and user filtering.

Existing event detection methods are vulnerable whilst filtering the events and ranking the popularity, which greatly reduces the efficiency and accuracy of event detection and event evolution. To resolve this problem, an automatic event and user scoring method [19], encompassing the authority value [7] and minimum distance of the posts, is considered essential. This method can be achieved based on the idea that the representative posts under specific event usually have higher authority and there is often an event difference evident between the posts in a microblogging network. This permits automatic filtering of events and event popularity ranking based on the event and user scoring method. Specifically, the HEE model assumes that the representative posts under specific event have the following features.

(1) Authority. Representative posts in microblogging network are surrounded by non-representative posts. Similar to the clustering method based on prototype, each topic is regulated by an event leading post. The event-lead posts have higher influence over their respective topic, which also reflects their higher authority. Thus, a post with high authority is more likely to be chosen as a representative topic of hot event.

(2) Topic scatters distribution. The representative posts' topic and users who publish them are different from each other in microblogging network. As each event is characterized by a representative post and key user, the representative posts and key users are expected to be in an evenly distributed fashion in microblogging network. Thus, a representative post should have a larger distance separation from other posts with different topic types.

To capture the representative posts characterized by the above two properties, an event and user scoring method is introduced, where one dimension shows the 'authority' properties of representative posts and the other dimension evaluates the 'topic scatters distribution' properties of the representative posts. Therefore, *K* posts located in the right upper hand of the picture can be automatically selected as representative posts. Then, the LDA topic model can be used to cluster posts in microblogging network into various events. For achieving the above process, the introduced topic scoring method needs a method to calculate minimum distance for describing the scatters distribution between representative posts under specific events.

In this paper the minimum distance of representative posts with other posts of higher authority is calculated. With the carefully selected topics, posts in microblogging network are clustered by LDA and Gibbs sampling [20-21].

The minimum distance δ_i *(i = 1, 2,…, n)* can be achieved by calculating the distance between post p_i and other posts with higher authority,

$$\xi_i = \min_{j:A_j > A_i}(d_{ij}) \tag{2}$$

where d_{ij} is the Euclidean distance between post P_i and post P_j.

In order to compute the Euclidean distance, we first need to transform the *n* posts into vectors within the same domain space. Different similarity methods [22] such as Jaccard similarity [23] and signal similarity [24] can be used to determine pairwise similarities. In this study, the similarity of the posts is calculated based on the signal similarity, since the signal similarity outperforms the Jaccard similarity according to the results of our experiments.

The signal propagating process is the core of signal similarity calculation. Therein, all posts will be taken as an initial signal source to send their signals to themselves and their neighbours each time respectively. In these processes, posts and all neighbours need record the amounts of signals they receive. After *t* steps, the amount of signals of posts can be regarded as the effect of this post on the other posts. Then the i^{th} column of *INF* indicates the effect of post P_i on the other posts in *t* steps. Based on this, we can obtain *n* vectors INF_1 , INF_2 ,…, INF_3 in Euclidean space. The process can be described as

$$INF = (A + I)^t \tag{3}$$

where *I* is an n-dimensional matrix, *t* is the total steps in signal propagating process, due to the fact microblogging networks are very sparse, the computation of *INF* is not time-consuming when we set the value of *t* is 3.

Based on the above formulas, the similarity of all posts can be calculated in the post network. It is noticeable that the authority value of posts is changed by the distance of the corresponding post from other posts in the post network, which indicates that the post located closer to other posts in the network will achieve larger authority value. This suggests that such posts are pretty influential than the surrounding posts.

Where posts with the same authority value exist, post which has smaller *P.id* ranks higher. If a post P_k with maximal authority value, A_k = max (A_i), is the most

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2017.2675839, IEEE Access

exceptional in comparison to its neighbours and is therefore a representative post. Thus, we define its minimum distance δ_k as:

$$\xi_k = \max(\xi_i), i \neq k \tag{4}$$

Using this minimum distance, there is distinct variation among different posts; high-quality posts in the microblogging network can be easily recognised by this rank value. Meanwhile, based on the assumptions of our model, the representative posts are those having higher authority values and those located dispersedly in the microblogging network. We use the event and user scoring method in 2-dimensional space to automatically filter the events and score the popularity of events, where one dimension is the authority value of the posts; another is the minimum distance of the posts as mentioned above. In this topic scoring method, posts located in the right upper co-ordinates are considered as the representative posts. Finally, filtering the events and users, and scoring the popularity of events can be completed according to the representative posts.

C. An Automatic Event Clustering Algorithm

1) The post weights

Suppose $E = \{E_1, E_2, \ldots, E_K\}$ is a event clusters of a graph $G(V, E)$, where V is the set of posts and E is the set of edges between two posts. The N posts in the graph can be denoted by $\{P_1, P_2, \ldots, P_N\}$. The matrix $V_{K\times N}$ denotes the post weights of N posts related to all the K events. As analysed previously, the authority of post can be used to express the opinion the post plays the centre role under its topic. Hence, the weight of post j's influential degree in topic cluster E_r can be described as:

$$\text{if } P_j \in E_r \; V_{rj} = \frac{Authority_r(j)}{\sum_{(h:P_h \in E_r)} Authority_r(h)} \quad \text{else } V_{rj} = 0 \; (r = 1,2,...,k,\ j = 1,2,...,N) \tag{5}$$

Therefore, for a given post P_i, the similarity between P_i and event E_j, described as, $\bar{s}_{ij}$ can be calculated as

$$\bar{s}_{ij} = \sum_{h=1}^{N} v_{jh} \times s_{jh} \tag{6}$$

where s_{jh} is the similarity between posts P_i and P_h. As we can see from function (5) and (6), $\bar{s}_{ij}$ is a sum of the similarity between post P_i and other posts in event E_j, thus the weights mainly rely on the contribution of the posts to the event.

2) The event automatic clustering algorithm

The algorithm for clustering posts into events in Microblogging networks is described as Algorithm 1[5].

Algorithm 1: The event automatic clustering algorithm

Input: K, the number of events; A, the link matrix; $Nmax$, the maximum number of iterations.

Initialization:

(1) Select the top K posts with the highest authority values for the initial K events.

(2) Calculate the similarity matrix between any two posts in the graph.

(3) Extract the similarity matrix between the posts and the events. Partition the post into the event to which its nearest event belongs, and get the initial K classes of the graph: $E_1, E_2, \cdots, E_K$.

Repeat

(4) Update the matrices $V_{K\times N}$ recording post weights of N post with respect to all the K events based on the current partitions using function (5).

(5) Calculate the similarity between post P_i and event E_j, $\bar{s}_{ij}$, using function (6), and then cluster the vertices into K events with every post being in the event it is most similar to.

Until: All the clustered events remain unchanged or the number of iterations comes to $Nmax$.

Output: All the members in each event.

D. The Community Attributes Extraction

The HEE model is also composed of two modules: First, with the scored topics, posts in the post network are clustered by LDA topic model and Gibbs sampling. Second, the user topic communities can be discovered through Multi-Prototype user topic Community Detection Method. Fig. 2 illustrates the process for extracting the user community attributes.

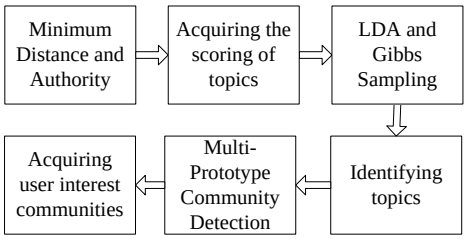


Figure 2. Procedure for Extracting the Community Attributes.

1) The user weights

Suppose $C = \{C_1, C_2, \ldots, C_K\}$ is a user community of a graph $G(V, E)$, where V is the set of users and E is the set of edges between users. The N users in the graph can be denoted by $\{u_1, u_2, \ldots, u_N\}$. The matrix $V_{K\times N}$ denotes the user weights of N users related to all the K user communities. As analysed previously, the hub of a user can be used to express the opinion the user plays the centre role under its community. Hence, the weight of user j's influential degree in community C_r can be described as:

$$\text{if } u_j \in c_r \qquad V_{rj} = \frac{Hub_r(j)}{\sum_{(h:n_h \in c_r)} Hub_r(h)} \quad \text{else } V_{rj} = 0 \; (r = 1,2,...,k,\ j = 1,2,...,N) \tag{7}$$

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2017.2675839, IEEE Access

Therefore, for a user u_i, the similarity between u_i and community C_j, described as, $\bar{s}_{ij}$ can be calculated as

$$\bar{s}_{ij} = \sum_{h=1}^{N} v_{jh} \times s_{jh} \quad (8)$$

where s_{jh} is the similarity between users u_i and u_h. As we can see from function (7) and (8), $\bar{s}_{ij}$ is a sum of the similarity between between user u_i and other users in community c_j, thus the weights mainly rely on the contribution of the users to the community they belong to.

2) The user community detection algorithm

The algorithm for clustering users into communities in Microblogging networks is described as Algorithm 2[5].

Algorithm 2: The user community detection algorithm

Input: *K*, the number of user communities; *A*, the link matrix; *Nmax*, the maximum number of iterations.

Initialization:

(1) Select the top *K* users with highest hub value [7, 26] as the initial *K* communities.

(2) Calculate the similarity between any two users in microblogging networks.

(3) Extract the similarity matrix between the users and the communities. Partition the user into the communities to which its nearest community belongs, and get the initial *K* classes of the graph: $C_1, C_2, \cdots, C_K$.

Repeat

(4) Update the matrices $V_{K\times N}$ recording prototype weights of *N* post with respect to all the *K* topics based on the current partitions using function (7).

(5) Calculate the similarity between user U_i and community C_j, $\bar{s}_{ij}$, using function (8), and then cluster the vertices into *K* user communities with every user being in the community it is most similar to.

Until: All the detected communities remain unchanged or the number of iterations comes to *Nmax*.

Output: All the members in each community.

E. The Cosine Measure and User Interest Discovering

After the completion of the LDA training process, the estimated *P(z)* parameter is used to find an important event under a topic z. Posts related to the topic *z* are sorted according to *P(d|z)* in descending order, and words related to the topic *z* are sorted according to *P(w|z)* in descending order. But it is difficult for each topic to judge whether they belong to the same real-life event. Existing methods identify new real-life events manually and subjectively. In the HEE model, a new method based on cosine measure is presented to judge whether a new hot event is emerging and to identify whether some topics are belonging to one event automatically.

Ground Truth. We manually choose 10 input events from the previous mentioned 100 events. These events all have obvious evolution processes in the real world and we can find out corresponding news reports of each development phase for them. To generate a baseline event evolution process, we choose a news title as the description for each phase and manually combine them into an evolution chain for each event.

Baseline. Some other works use the combination of multiple features to measure the relationships between two events [29-30]. Content similarity and temporal proximity are most widely used and proved to be significant on news corpora [30]. Therefore we adopt the combination of content similarity and temporal proximity as the baseline.

1) Cosine similarity measure

It is difficult to analyse and understand the information obtained from microblogging networks because the posts generated by users are mostly short text, camouflaged within a variety of topics and are not published in strict time sequence. To resolve this issue, a long document is composed from posts in each of the topic communities to determine a number of key keywords in each topic that represent the major event. The overall topic in each document is similar. Because the document topic is similar, the word composition is more likely to belong to the same topic. This solves the problem of short text feature sparsity and strengthens the ability to learn within the topic which improves the quality of the topic analysis. At the same time, in the process of learning the topic, the topic direction of the user can be obtained by the topic direction of the document.

The posts are re-processed by eliminating the stop list and by word segmentation. Each post d_i is a set of individual words without keyword extraction and can be expressed as a vector. The representative post d_i in each event is chosen as the major event and key post d_k. The included angle cosine is used as the standard to judge correlation between posts and the key post, d_k. The cosine distances between posts d_i and key post d_k are calculated as follows (9):

$$\cos(d_i, d_k) = \frac{\vec{d}_i \cdot \vec{d}_k}{\|d_i\| \|d_k\|} \quad (9)$$

Higher values of $cos(d_i,d_k)$ indicate a higher event similarity. A threshold, λ_p, is defined and the post d_i and key post d_k are considered to be related to the same event when $cos(d_i,d_k)$ of d_i is greater than λ_p. In addition, a share η is needed and calculated in (10).

$$\eta = \frac{n(d_k)}{r} \quad (10)$$

where $n(d_k)$ denotes the number of posts similar to standard post d_k. The threshold λ_p is confirmed by training the sample topics marked with fixed η.

2) Personalized User Interest Discovery Architecture

To be able to effectively obtain the interests of a given user in the HEE model who has published, forwarded or replied to very few posts, a collaborative filtering based user interest discovery method is proposed, named UID-LDA. This method takes full advantage of latent user profile based collaborative filtering and explicit user profile based

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2017.2675839, IEEE Access

collaborative filtering. It also considers the importance of topic key words on user interest generation. The architecture of the personalized user interest discovering is shown in Fig. 3.

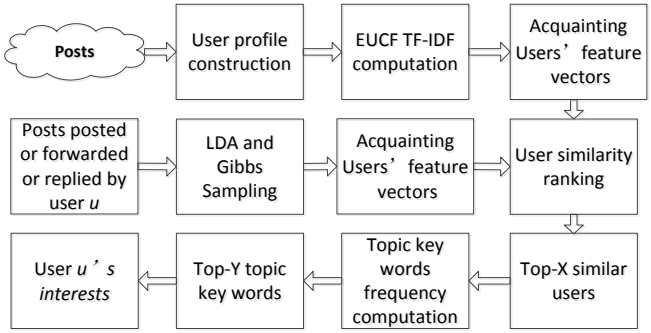


Figure 3. The Procedure of personalized user interest discovering method

Microblogs posted by user *u* construct user profile of *u*. Before obtaining the users' interests, parameters should be estimated from the training data. The UID-LDA method is used to construct users' feature vectors, constructing users' feature vectors by a TFIDF [25] schema.

With the feature vectors computed, our proposed approach can obtain the interests of a certain user. Specifically, when a user *u* posts, forwards or replies to a post, the UID-LDA method can compute the similarity between user *u* and other users individually using the cosine similarity measure. The measure is then used to select *u*'s X-nearest, or most similar, neighbours.

After the top-X similar users are obtained, UID-LDA method computes key words that appear in the top-X similar users' user profiles. These key words are ranked to their frequencies and the top-Y frequent words are discovered for user *u* as user *u*'s interest.

3) Algorithm of personalized user interests discovery

Kywe et al. [27] proposed an explicit user profile based collaborative filtering method, EUCF. This method represents each user by a vector. The weight of the vector is calculated by a TF-IDF schema where the TF is frequency of key word in user *u*'s user profile and IDF is the number of users who have used the key words. After the weights of all users are computed, the method recommends topics. Although EUCF performs well, it has a significant functional limitation: as EUCF finds similar users based TF-IDF schema, the results tend to be local.

To remedy this defect, we proposed a user interest discovery based LDA model, named UID-LDA. With UID-LDA, we can first find latent factors in the whole dataset and then represent each user by a feature vector. The vectors are the user-interest distributions (i.e. parameter *u*) estimated. As UT-LDA mines the latent relationships between users, interest and words, the similar users found by this method tend to be global. Moreover, UID-LDA also takes full advantage of the importance of key words when mining the latent relationships. The dimension of the vector (i.e. value of parameter *T*) is relatively low.

Given a target user *A* and another user *B*, let Similarity (*A*, *B*) denote distance between *A* and *B* which is computed by EUCF. Algorithm 3 shows the personalized user interests discovery algorithm.

Algorithm 3: personalized user interests discovery

Input: *P*, the number of posts posted by user *u*; *U*, the number of similar users; *H*, the number of topic key words.

Initialization:

(1) Select the top *K* users with highest hub value [7, 26] as the initial *K* interests.

(2) Construct user profile feature vector *F*.

(3) Calculate Similarity (*A*,*B*) according to formula (9) between any two users in the user community.

(4) Select the top-X similar users according to Similarity (*A*,*B*).

(5) Compute frequencies of the topic key words that appear in the top-X similar users' user profile.

(6) *T*= set of the top-Y frequent interest key words.

Output: *T*.

IV. EXPERIMENTS

In this section, we detail the experiments we conducted on a real-world short-text collections in order to demonstrate the efficiency and accuracy of our proposed HEE method. We consider four typical topic models as our benchmark methods, namely PLSA, LDA and EVE [7].

And the rest of this section describes the collection of dataset, experimental setup and analysis, the baseline approach and the model evaluation.

A. Dataset

Our dataset are collected from Twitter [28] (http://twitter.com/) via Twitter API. The collected dataset is composed of 126995 posts and 6589 users from December 28, 2015 to January 05, 2016.

B. Experimental Settings

The experiments were conducted on a machine with Intel I3 3.4 GHz CPU and 4G memory.

Parameters were tuned via grid search. For PLSA, BEE and EVE, the mixture weight of the background model λ_B was fixed to 0.05 [8]. For LDA, α = 0.5 and β = 0.1. In all the methods, Gibbs sampling was run for 1,000 iterations; EM algorithm was also run for 1,000 iterations. The results reported are the average over 5 runs. In the process of filtering high-quality posts and high-influence users, all of the initial authority scores *d.a* and *u.h* hub scores were set to 1.

C. Baseline Approaches

The efficiency and effectiveness of the proposed HEE is validated by evaluating our model against PLSA, LDA, and EVE [7], which are the classic latent semantic analysis algorithms.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2017.2675839, IEEE Access

D. Evaluation Methods

1) Tweets and users filtering: As previously discussed, we have explored HITS-based event and user filtering approaches [7,19] for hot events and high-influence users' filtering. The results are shown in Table I. Since most tweets are not event-related, it makes sense to only report the results on the event-related class. The performance obtained here is comparable to the state-of-the-art results on tweet. We found that the key posts-based approach gives higher precision compared to the keyword-based approach. As such, we chose to use the key posts-based approach for tweets and users filtering in the experiments. After the filtering step, we are left with hot events and high-influence users. These events and users are used for event extraction and categorization. It can be observed that the filtering step is really crucial to event extraction. By filtering out general events and low-influence users, the precision of our event extraction component increases dramatically from 66.58% to 83.75% in Table V. When compared against the baseline approach, EVE model, it can be observed from Table VII that our proposed framework significantly outperforms the baseline with nearly 57.76% improvement on efficiency. This is because EVE model only deal with tweets for filtering the low-quality tweets, ignoring the filtration of general events and low-influence users. Meanwhile, another possible reason is that LDA's Gibbs Sampling algorithm is faster than PLSA's EM algorithm in a large scale Twitter data which can be seen from the Table VII.

2) The process of filtering popular events: As is shown in Fig. 4, we can see the authority value and minimum distance of each post, which can distinguish the importance of posts and discover the key posts under hot events. Meanwhile, we can also discover the number of popular events that locate in the right upper quadrant in Fig. 4, and Fig. 5, which plays a key role to the spread of influence under a specific user interest community we choose. And it can be observed from Table I, our proposed HEE method can detect the proper top k (k is set to 15 in Table I and Table II) high-quality posts according to their authority value efficiently and effectively, with which we can find the proper number of popular topics. In addition, when the authority value of posts is equal, it can be sorted according to the minimum distance of posts, which show us the key posts under popular topics. At the same time, it can be seen from Table III, the proposed HEE method can detect the number of popular events under the different parameter *k*, and by setting different parameter *k*, and the proper number of popular events can also be discovered. Meanwhile, the key users and the key posts related to these hot events propagation can be explored from Table IV, which further improves the efficiency of event detection.

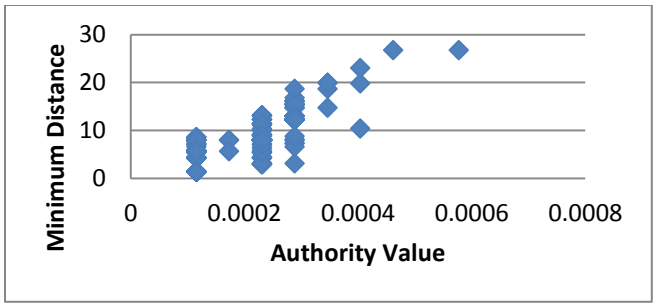


Figure 4. The number of topics from the HEE method

TABLE I. MINIMUM DISTANCE AND AUTHORITY OF POSTS

Key Post ID	Authority Value	Minimum Distance
681695337304702976	0.000461734	26.73948391
681693469564383232	0.000577167	26.73948391
681697033523077122	0.000404017	23
681696678097719296	0.0003463	19.94993734
681697799730249728	0.0003463	19.92485885
681699684168015873	0.0003463	18.70828693
681994493239803904	0.000288584	18.68154169
681695517328277505	0.000288584	16.85229955
681353451692011520	0.000404017	15.93737745
681262717555085313	0.000577167	15.93737745
681353050188070912	0.000288584	11.04536102
684205547067944961	0.000230867	10
681352883275718656	0.000288584	7.211102551
684205255907643392	0.000230867	7.141428429
681353806282649602	0.000288584	6.557438524

TABLE II. KEY POSTS UNDER POPULAR TOPICS

Key Post ID	Popular Topic
681695337304702976	Life
681693469564383232	Basketball
681697033523077122	Basketball
681696678097719296	Basketball
681697799730249728	Basketball
681699684168015873	Basketball
681994493239803904	Life
681695517328277505	Basketball
681353451692011520	Business
681262717555085313	Conflict
681353050188070912	Business
684205547067944961	Economy
681352883275718656	Business
684205255907643392	Economy
681353806282649602	Business

TABLE III. RESULT OF DIFFERENT PARAMETER *K*

Method	Number of popular topics under the different *k*						
	k=1	*k=2*	*k=5*	*k=8*	*k=10*	*k=15*	*k=20*
HEE	1	2	2	3	4	5	5

TABLE IV. EVALUATION RESULTS FOR EVENT DETECTION WITH HOT EVENT FILTERING

Minimum distance	Name entities	Key posts	Created Time

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2017.2675839, IEEE Access

26.73948391	Unneededbody	@Unneededbody yeah, the both of you would be better off without me	Tue Dec 29 04:36:48 +0000 2015
26.73948391	basketball_ne, damonbenning, GarySharp1620, NickBahe, neb	@basketball_ne @damonbenning @GarySharp1620 @NickBahe if you were a div 1 caliber player from Nebraska, would u play @ neb? Zero ncaa wins	Tue Dec 29 04:29:23 +0000 2015
23	basketball_ne, damonbenning, GarySharp1620, NickBahe, bigKuxy	@basketball_ne @damonbenning @bigKuxy @GarySharp1620 @NickBahe Jeff Riley from GICC and even Matt Davison were that year.	Tue Dec 29 04:43:32 +0000 2015
19.94993734	basketball_ne, GarySharp1620, NickBahe	@basketball_ne @GarySharp1620 @NickBahe Crazy run..don't forget Pugh & Alton Mason. 2 that away	Tue Dec 29 04:42:08 +0000 2015
19.92485885	basketball_ne, damonbenning, GarySharp1620, NickBahe	@damonbenning @basketball_ne @GarySharp1620 @NickBahe smaller scale but Wesleyan had national champion basketball team with all local kids	Tue Dec 29 04:46:35 +0000 2015
18.70828693	NWU42, damonbenning, GarySharp1620, NickBahe, 1344JB	@damonbenning @1344JB @GarySharp1620 @NickBahe @NWU42 Well there is a guy who could play! Dana Janssen for NWU.	Tue Dec 29 04:54:04 +0000 2015
18.68154169	mysavage	@mysavage_life Lmfaooo that was like a week ago □ time to go again	Wed Dec 30 00:25:32 +0000 2015
16.85229955	ChasersNUJays, basketball_ne, GarySharp1620, NickBahe	@ChasersNUJays @basketball_ne @GarySharp1620 @NickBahe Not well, but your point is still irrelevant. In the mid 90's. Local kids picked NU	Tue Dec 29 04:37:31 +0000 2015
15.93737745	Gabe_Energy, MyranSSB	@Gabe_Energy @MyranSSB Power Rankings and Pro Points for what? Myran is more than likely going to be on the main FL PR when it gets updated	Mon Dec 28 05:58:16 +0000 2015
15.93737745	Yalelkm	OPINION: To defeat ISIL, the Israeli-Palestinian conflict must be resolved... https://t.co/E8VcCpb8MC https://t.co/3SfdtB6b2d	Sun Dec 27 23:57:43 +0000 2015
11.04536102	Gabe_Energy, MyranSSB	@Gabe_Energy It's between that and the cE one, cE one has low attendance and @MyranSSB wants to go to something worth the drive.	Mon Dec 28 05:56:40 +0000 2015
10	pilotolson	@pilotolson so what's the big risk for us? Where do you see the US economy in 10 or 30 years?.	Tue Jan 05 02:51:29 +0000 2016
7.211102551	ImChemX	@ImChemX nah the cE one	Mon Dec 28 05:56:00 +0000 2015
7.141428429	fr_christopher	@fr_christopher ...a stage that every country goes through I'm development, just like the US and every other so-called "advanced" economy.	Tue Jan 05 02:50:19 +0000 2016
6.557438524	ImChemX MyranSSB	@ImChemX @MyranSSB Its going to be a national ranking not just Florida	Mon Dec 28 05:59:41 +0000 2015

3) The event similarity analysis: As is shown in Table V, we can easily discover the similar events during event evolution based on the HEE model. This is because a cosine measure based event similarity detection method is proposed to judge correlation between events, which can detect the process of event evolution. At the same time, we can also find which hot events have a long event evolution chain, which indicates the popularity of hot events compared with the EVE model [7]. Meanwhile, with the improved LDA topic model, we can detect the interests of some key users from Table V, which addresses the data sparsity due to the short text of posts and increases the accuracy of user interest community detection.

TABLE V. PROCESS OF EVENT EVOLUTION

Time	***Hot Event***	***Topic***	***Key posts id***	***Event evolution relationship***
Sun Dec 27 2015	E1	Conflict	681262717555085313	
Mon Dec 28 2015	E2	Business	681352883275718656	
	E3	Business	681353050188070912	E2→E3
	E4	Business	681353451692011520	E2→E3→E4
	E5	Business	681353806282649602	E2→E3→E4→E5
Tue Dec 29 2015	E7	Life	681693469564383232	
	E8	Basketball	681695337304702976	
	E9	Basketball	681695517328277505	E8→E9

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2017.2675839, IEEE Access

Time	*Hot Event*	*Topic*	*Key posts id*	*Event evolution relationship*
	E10	Basketball	681696678097719296	E8→E9→E10
	E11	Basketball	681697033523077122	E8→E9→E10→E11
	E12	Basketball	681697799730249728	E8→E9→E10→E11→E12
	E13	Basketball	681699684168015873	E8→E9→E10→E11→E12→E13
Wed Dec 30 2015	E14	Life	681994493239803904	E7→E14
Tue Jan 05 2016	E15	Economy	684205255907643392	
	E16	Economy	684205547067944961	E15→E16

TABLE VI. THE RESULTS OF KEY USERS' INTERESTS DETECTION

Key user	*Interest words in T_{k-1}*	*Interest words in T_k*	*Interest words in T_{k+1}*
basketball_ne	Basketball	Life	
damonbenning	Basketball		
GarySharp1620	Basketball		
NickBahe	Basketball	Business	
Gabe_Energy,	Economy		Business
Yalelkm	Conflict	Life	
MyranSSB	Economy		
pilotolson	Economy		Basketball
ImChemX	Economy		
fr_christopher	Economy		Basketball

TABLE VII. COMPARISON OF TIME EFFICIENCY

Method	**Time (K=15)**				
	HITS [26]	*Topic decision method*	*Gibbs sampling*	*EM*	*Total*
PLSA	0	0	0	36.35min	36.35min
LDA	0	0	23.56min	0	23.56min
EVE	28862 ms	0	0	24.12min	24.6min
HEE	28862 ms	6.37min	3.54min	0	10.39min

4) The results of key user interest discovering and changing: It is seen from the Table VI and Table VIII; we can discover that the key users' interests will be changed over time, which validates our proposed HEE model in detecting more interests for each user in users' community, while solves the big problem of data sparsity in microblogging networks. At the same time, with our proposed automatic topic clustering algorithm, all short texts can be combined into clusters with similar topics. And then with the improved user-interest model all short texts in each cluster can be integrated to form a long text document simplifying the determination of the overall topic in relation to the interest distribution of each user during the evolution of hot events. This addresses the problem of sparse data and improves the quality of topic definition and the accuracy of user interest discovering.

TABLE VIII. THE RESULTS OF PERSONALIZED USER INTEREST DISCOVERING METHOD

Key User	*Interests*		
basketball_ne	Basketball	Life	Business
damonbenning	Basketball	Life	Business
GarySharp1620	Basketball	Life	Conflict
NickBahe	Basketball	Business	Life
Gabe_Energy,	Economy	Basketball	Business
Yalelkm	Conflict	Basketball	Business
MyranSSB	Economy	Basketball	Business
pilotolson	Economy	Life	Basketball
ImChemX	Economy	Basketball	Conflict
fr_christopher	Economy	Life	Basketball

V. CONCLUSION AND FUTURE WORK

Microblogging is a form of social media which allows people to share and disseminate real-life events. However, many existing approaches primarily discuss event detection model, but ignore the user interest discovering during event evolution with the problem of low efficiency and low accuracy posed by the fact that the influential users' interests will change during the event evolution. To address this problem, we propose a user-interest model based event evolution model, named HEE model. This model not only considers the user's interest distribution, but also uses the short text data in the social network to model the posts and applies the recommend methods to discovering the users' interest, which can solve the problem of data sparsity existed in many existing event detection methods, and also improve the accuracy of event detection. Finally, the experimental results on real Twitter dataset demonstrate the efficiency and accuracy of our proposed model for both event evolution and user interest discovering.

In the future work, we intend to consider other attributes of tweets (e.g., embedded URL) to compute the relatedness between two tweets for customizing the concepts co-occurrences within a single tweet to an increased. Therefore we plan to apply other types of community detection methods to better extract posy topics. Meanwhile, the next research points are also including about how to predict the behaviour changes of influential users during the evolution and how to predict the popularity of hot events in the future.

ACKNOWLEDGEMENT

This work was partially supported by the National Natural Science Foundation of China under Grants No. 61502209 and 61502207.

REFERENCES

[1] Q. Gao, F. Abel, G.J. Houben, et al., A comparative study of users' microblogging behavior on Sina Weibo and Twitter, in: User Modeling, Adaptation, and Personalization, Springer, Berlin, Heidelberg, 2012, pp. 88-101.

[2] Q. Diao, J. Jiang, F. Zhu, and E.-P. Lim, "Finding bursty topics from microblogs," in Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics: Long Papers-Volume 1, 2012, pp. 536-544.

[3] X. Zhou and L. Chen, "Event detection over twitter social media streams," The VLDB Journal—The International Journal on Very Large Data Bases, vol. 23, pp. 381-400, 2014.

[4] D. M. Blei, A. Y. Ng, and M. I. Jordan, "Latent dirichlet allocation," Journal of machine Learning research, vol. 3, May. 2003, pp. 993-1022.

[5] K. Zhou, A. Martin, and Q. Pan, "A similarity-based community detection method with multiple prototype representation," Physica A: Statistical Mechanics and its Applications, vol. 438, pp. 519-531, 2015.

[6] T. Hofmann, "Probabilistic latent semantic indexing," in Proceedings of the 22nd annual international ACM SIGIR conference on Research and development in information retrieval,8. 1999, pp. 50-57, doi>10.1145/312624.312649 .

[7] X. Sun, Y. Wu, L. Liu, and J. Panneerselvam, "Efficient Event Detection in Social Media Data Streams," in IEEE International Conference on Computer and Information Technology; Ubiquitous Computing and Communications; Dependable, Autonomic and Secure Computing; Pervasive Intelligence and Computing, 2015.

[8] J.X. Li, Z.Y. Tai, RC. Zhang and W.R. Yu. "Online bursty event detection from microblog," in Proceedings of the 2014 IEEE/ACM 7th International Conference on Utility and Cloud Computing.IEEE Computer Society, 2014, pp. 865-870, doi:10.1109/UCC.2014.141.

[9] G. Salton, A. Wong, and C. S. Yang, "A vector space model for automatic indexing," Communications of the Acm, vol. 18, pp. 273-280, 1975.

[10] J. Makkonen, "Investigations on event evolution in TDT," in Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology: Proceedings of the Hlt-Naacl 2003 Student Research Workshop, 2004, pp. 43-48.

[11] R. Nallapati, A. Feng, F. Peng, and J. Allan, "Event threading within news topics," in Proceedings of the thirteenth ACM international conference on Information and knowledge management. ACM, 2004, pp. 446–453.

[12] O. Jin, N. N. Liu, K. Zhao, Y. Yu, and Q. Yang, "Transferring topical knowledge from auxiliary long texts for short text clustering," in ACM Conference on Information and Knowledge Management, CIKM 2011, Glasgow, United Kingdom, October, 2011, pp. 775-784.

[13] Y. Chen, X. Zhang, Z. Li, and J. P. Ng, "Search engine reinforced semi-supervised classification and graph-based summarization of microblogs," Neurocomputing, vol. 152, pp. 274-286, 2015.

[14] L. Hong and B. D. Davison, "Empirical Study of Topic Modeling in Twitter," Proceedings of the Sigkdd Workshop on Social Media Analytics, pp. 80-88, 2010.

[15] W. X. Zhao, J. Jiang, J. Weng, J. He, E. P. Lim, H. Yan, et al., "Comparing Twitter and Traditional Media Using Topic Models," Lecture Notes in Computer Science, vol. 6611/2011, pp. 338-349, 2011.

[16] X. Yan, J. Guo, Y. Lan, and X. Cheng, "A biterm topic model for short texts," in International Conference on World Wide Web, 2013, pp. 1445-1456.

[17] P. Yali, Y. Jian, L. Shaopeng, and L. Jing, "A Biterm-based Dirichlet Process Topic Model for Short Texts," 2014.

[18] M. Rosen-Zvi, T. Griffiths, M. Steyvers, and P. Smyth, "The author-topic model for authors and documents," in Conference on Uncertainty in Artificial Intelligence, 2012, pp. 487-494.

[19] Y. Li, C. Jia, and J. Yu, "A parameter-free community detection method based on centrality and dispersion of nodes in complex networks," Physica A: Statistical Mechanics and its Applications, vol. 438, pp. 321-334, 2015.

[20] A. Asuncion, M. Welling, P. Smyth, and Y. W. Teh, "On smoothing and inference for topic models," in Proceedings of the Twenty-Fifth Conference on Uncertainty in Artificial Intelligence, 2009, pp. 27-34.

[21] R. Alhamzawi and K. M. Yu, "Variable selection in quantile regression via Gibbs sampling," Journal Of Applied Statistics, vol. 39, pp. 799-813, 2012.

[22] L. Lü and T. Zhou, "Link prediction in complex networks: A survey," Physica A: Statistical Mechanics and its Applications, vol. 390, pp. 1150-1170, 2011.

[23] P. Jaccard, Etude comparative de la distribution florale dans une portion des Alpes et du Jura: Impr. Corbaz, 1901.

[24] Y. Hu, M. Li, P. Zhang, Y. Fan, and Z. Di, "Community detection by signaling on complex networks," Physical Review E, vol. 78, p. 016115, 2008.

[25] J.H. Paik, A novel TF-IDF weighting scheme for effective ranking, in: Proceedings of the 36th International ACM SIGIR Conference on Research and Development in Information Retrieval, ACM, 2013, pp. 343–352.

[26] S. Chakrabarti, B. Dom, P. Raghavan, S. Rajagopalan, D. Gibson, and J. Kleinberg, "Automatic resource compilation by analyzing hyperlink structure and associated text," Computer Networks and ISDN Systems, vol. 30, pp. 65-74, 1998.

[27] S.M. Kywe, T.-A. Hoang, E.-P. Lim, F. Zhu, On recommending hashtags in twitter networks, in: Proceedings of the 4th International Conference on Social Informatics, Springer, 2012, pp. 337–350.

[28] Twitter, REST API v1.1 Resources, 2014, Available at: https://dev.twitter.com/docs/api/1.1.

[29] D. Huang, S. Hu, Y. Cai, and H. Min, "Discovering event evolution graphs based on news articles relationships," in e-Business Engineering (ICEBE), 2014 IEEE 11th International Conference on. IEEE, 2014, pp. 246–251.

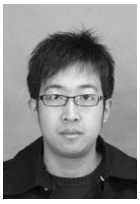
[30] R. Nallapati, A. Feng, F. Peng, and J. Allan, "Event threading within news topics," in Proceedings of the thirteenth ACM international conference on Information and knowledge management. ACM, 2004, pp. 446–453.

Lei-lei Shi received the B.S. degree from Nantong University, Nantong, China, in 2012, and the M.S. degree from Jiangsu University, Zhenjiang, China, in 2015. He is currently working towards the Ph.D. degree at the School of Computer Science and Telecommunication Engineering, Jiangsu University, Zhenjiang, China. His research interests include Event Detection, Data Mining, Social Computing and Cloud Computing.

Lu Liu is the Professor of Distributed Computing in the University of Derby, UK and adjunct professor in Jiangsu University,

China. Prof. Liu received his Ph.D. degree from University of Surrey and M.S. degree from Brunel University. Prof. Liu's research interests are in areas of Cloud Computing, Social Computing, Service oriented Computing and Peer-to-Peer Computing. He is the Fellow of British Computer Society (BCS).

Yan Wu received the M.S. degree from Shandong University of Science and Technology, Qingdao, China, in 2009, and the PhD degree from Tongji University, Shanghai, China, in 2014. He is currently a Lecturer with the School of Computer Science and Telecommunication Engineering in Jiangsu University, China. His research interests include formal methods, service-oriented Computing, and Cloud Computing.



Liang Jiang received the B.S. degree from Nanjing University of Posts and Telecommunications, China, in 2007, and the M.S. degree from Jiangsu University, Zhenjiang, China, in 2011. He is currently working towards the Ph.D. degree at the School of Computer Science and Telecommunication Engineering, Jiangsu University, Zhenjiang, China. His research interests include OSNs, Computer Networks and Network Security.

James Hardy is a PhD student in The University of Derby, UK. Besides manufacturer specific data networking qualifications, he holds a B.Eng. in Electronic Engineering from Nottingham University and a distinction level M.Sc. in Computer Networks from The University of Derby. He has industrial experience including aerospace control systems and global scale wide area data networking. Research interests include Smart City Transportation, Autonomous Vehicles, Communication Systems, Green Computing, HaaS, IaaS, control systems, simulation and virtualization.